Lecture 1

Introduction

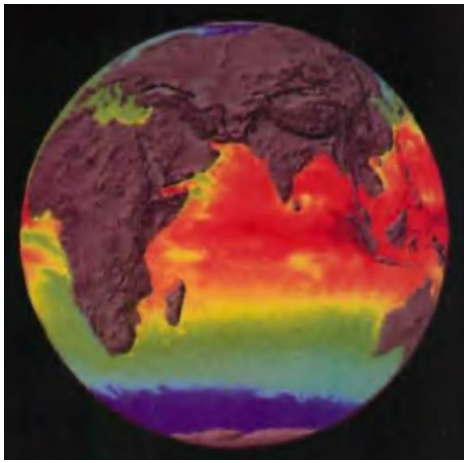
Computer graphics: Definition

- Computer graphics is concerning with all aspects of producing pictures or images using a computer.
- It specifically deals with images/pictures produced using programs
  - Writing a program in a general purpose language to produce an image
  - Using a graphics software package to draw an image
- This exclude that act of acquiring images using cameras (or other sensors) and manipulating it using the computer as this enter in the field of digital image processing

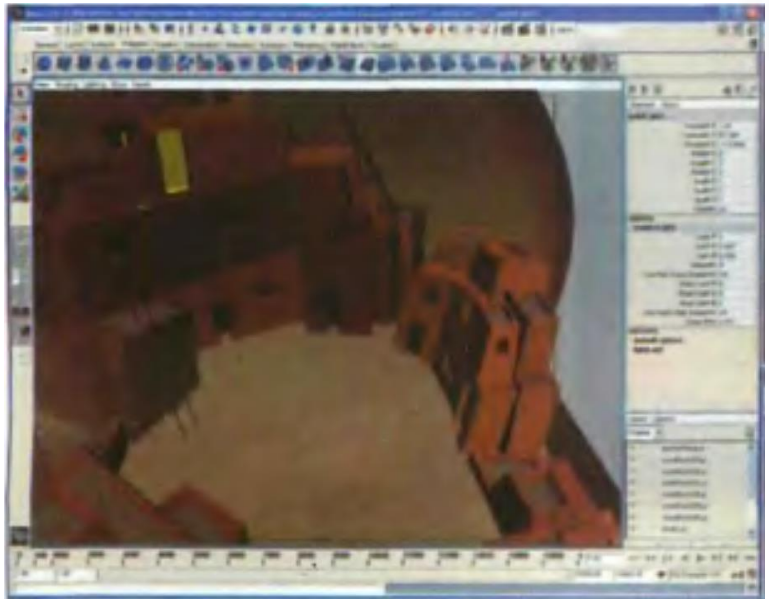
Application of computer graphics

- Display of information
  - The human visual system is unrivaled as a pattern recognizer
  - Aiding viewers in understanding information (Information visualization)
- Design(CADs systems)
- Simulation and modeling (ex. Graphical flight simulator: real time graphics production, games, VR)
- User interfaces

Application of computer graphics



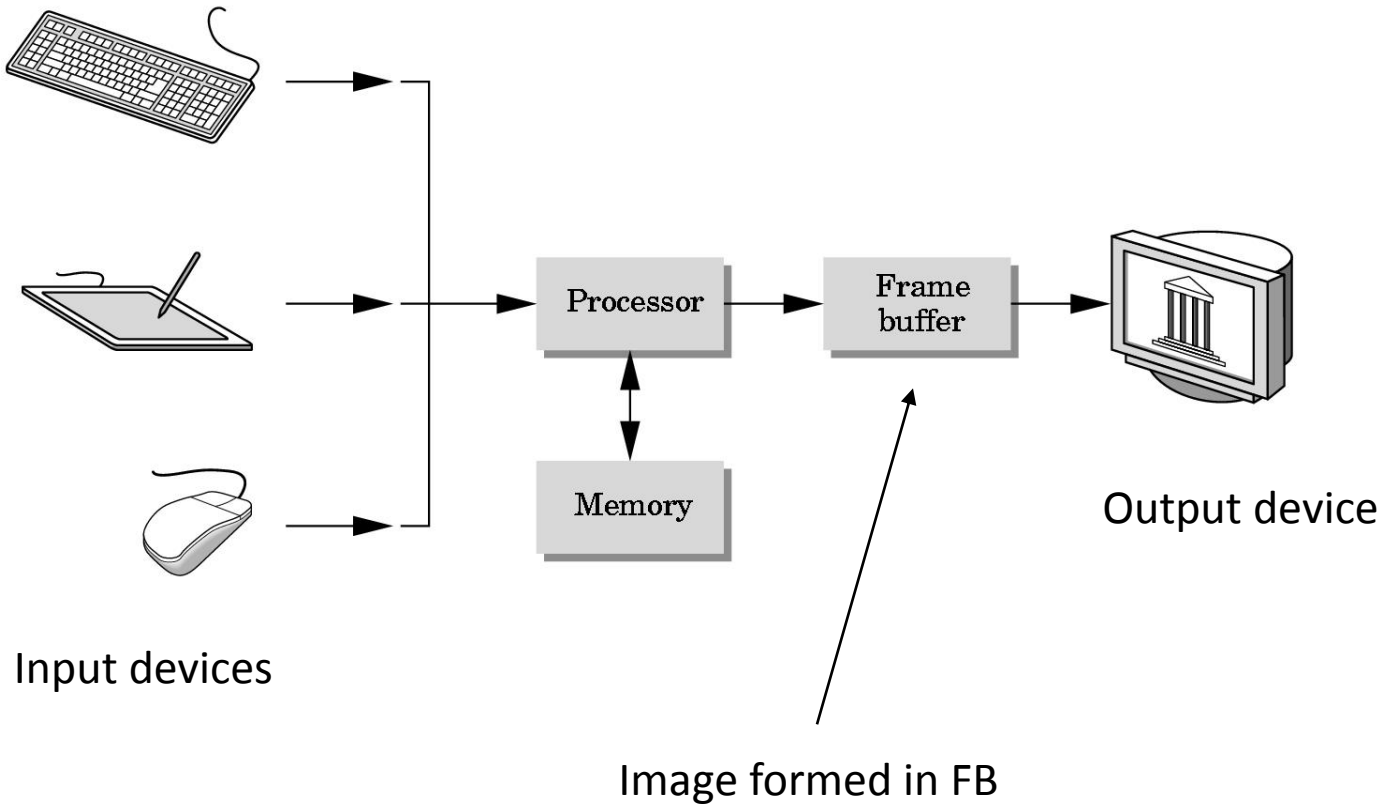
Information visualization: ox and temperature distribution



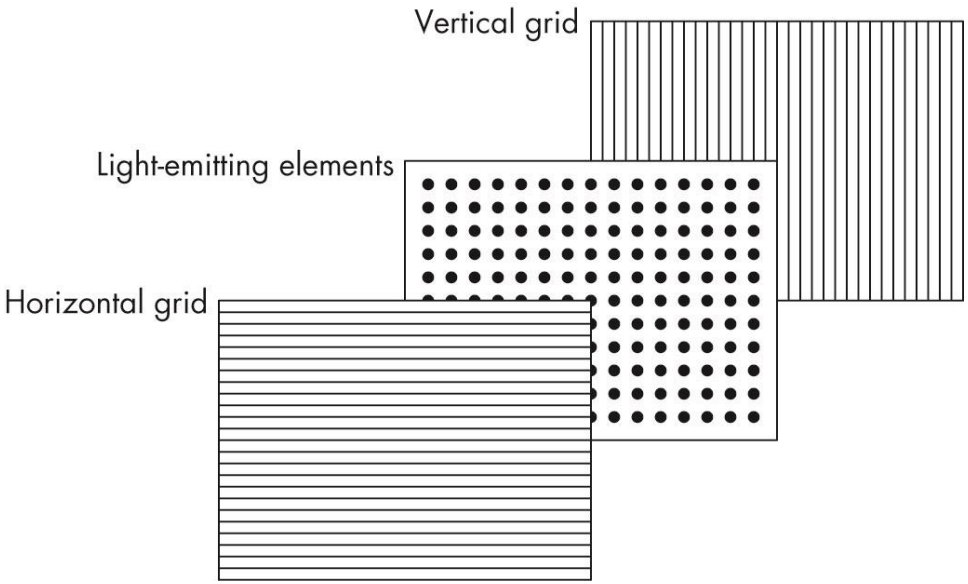
User interface: A huge number of tools presented simultaneously

Interactive design: Showing perspective view of a building

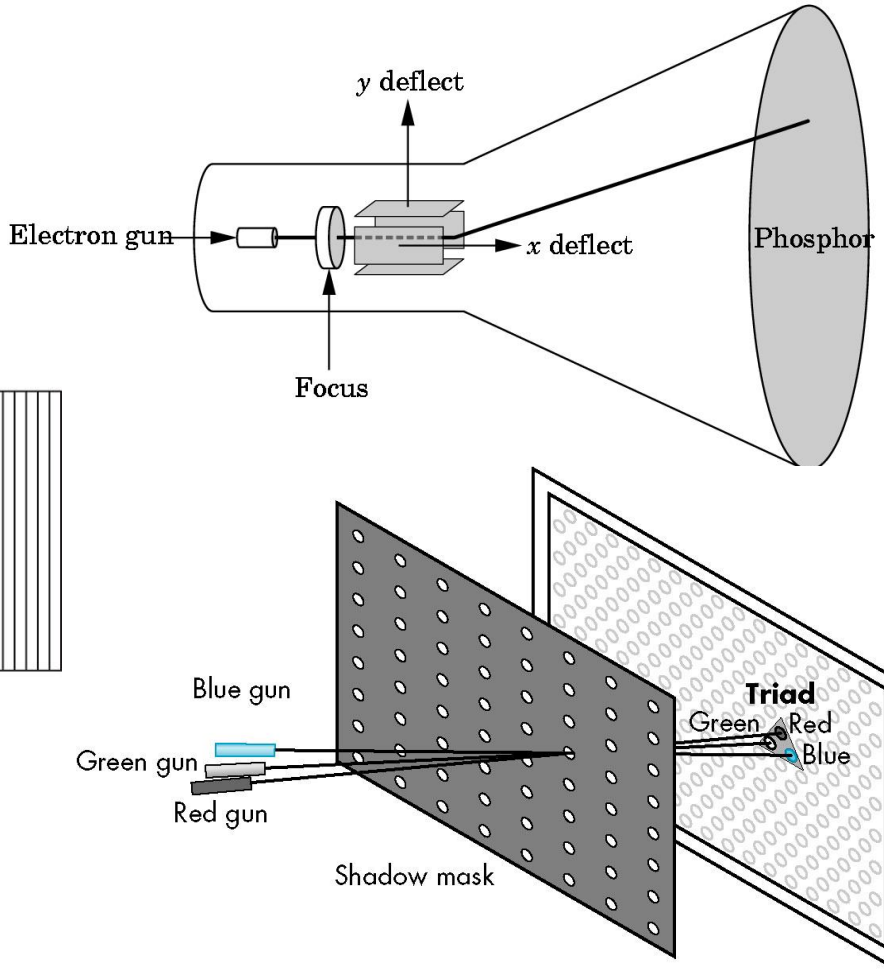
Graphics System



Output devices



Flat panel monitors are inherently raster. The vertical and horizontal lines specifies the element to be exited to emit light in the required color



CRT, and shadow-mask CRT. It Can be used either as a line-drawing device (calligraphic) or to display contents of frame buffer interlaced or progressively (raster mode)

Report Discussion

Previous lecture repot:
None

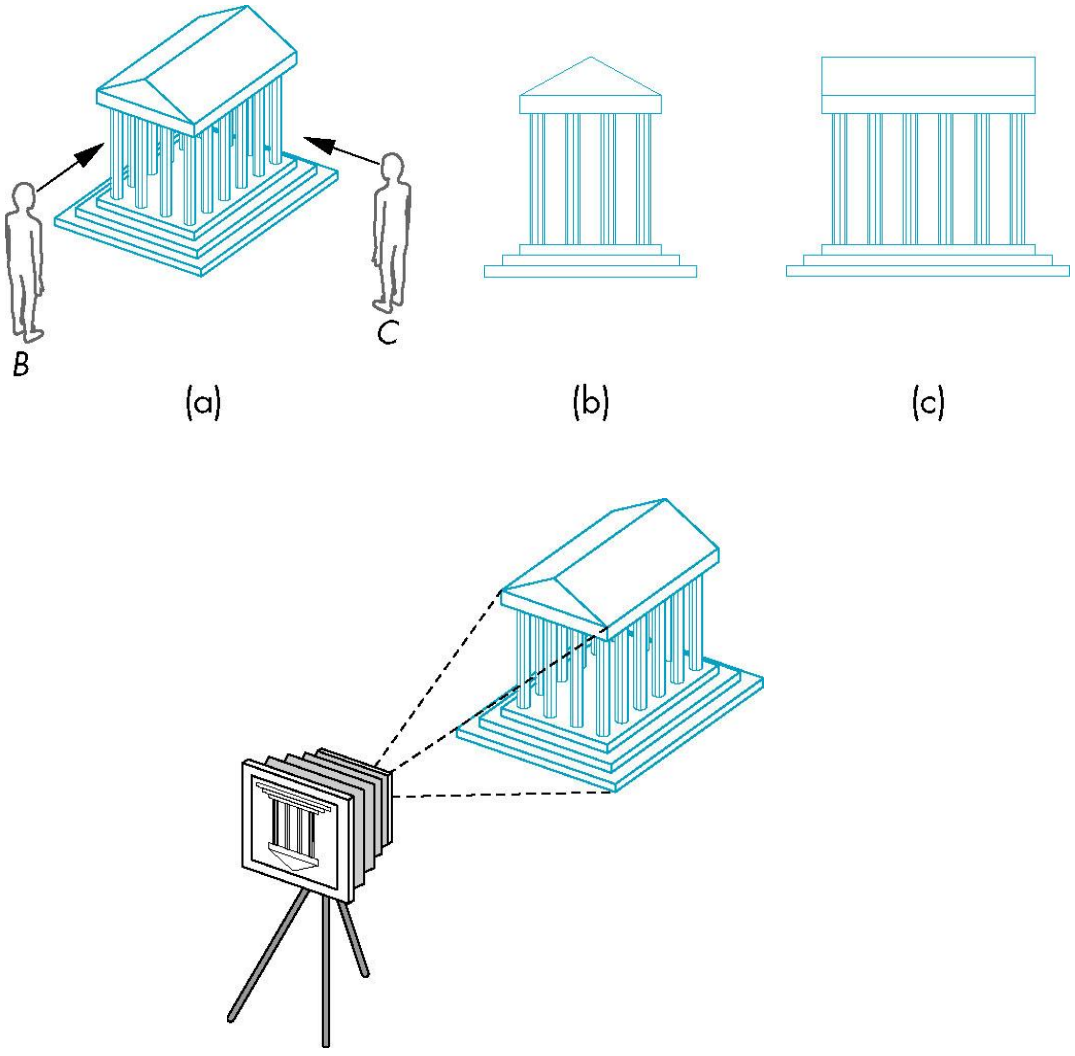
Next lecture repot:
Explain why Images are better displayed than text on CRT monitors?

Generating an image using the computer

- Forming image using the simple two dimensional geometrical entities (line, points, polygons)
  - Low level functions to rasterizing the 2D entities in FB
  - High-level function to reduce complex 3D into 2D
- Generating images through using a model for image generation that imitates optical imaging systems (cameras and human visual system)
  - Specifying what exists in the scene, where the light sources(s) is(are) located, what is the nature of the scene materials, etc
  - A special software (graphics library) and hardware (GPU) cooperates to produce the scene according to your specification by applying the imaging model

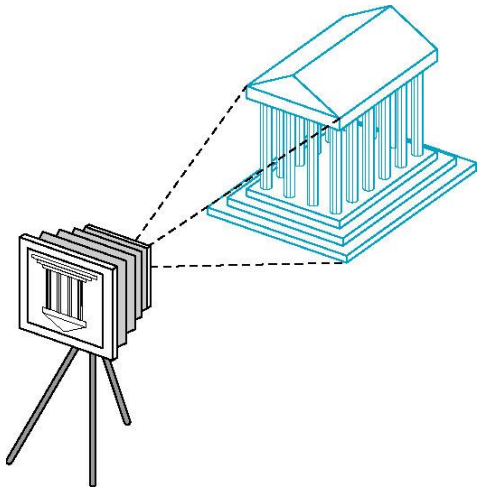
Imaging model: Objects and viewer

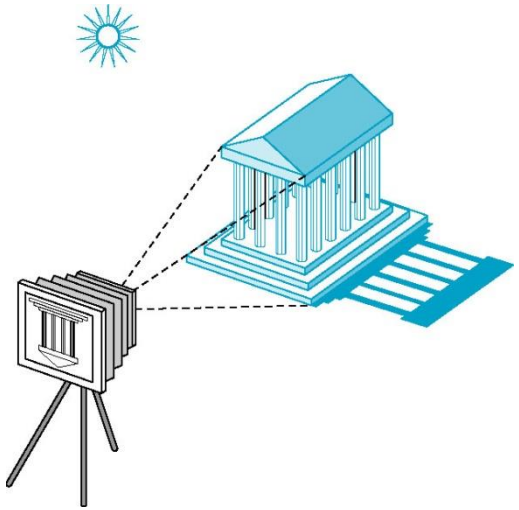
- Images are formed when a viewer looks at objects
- Objects are exist in the world (scene)
- We can specify objects in the scene using simple elements such as points, lines, and polygons. Usually these simple objects themselves are specified by points in a 3D coordinate system called vertices
- The resulting image depends on the location of the viewer with respect to the objects and the view direction of the viewer
- It's clear that the imaging model must take object and viewer into consideration



Imaging model: Light and images

- The previous model is not able to illustrate the role of light in images
  - Colors
  - Shining spots in the scene
  - Without any lights, we see nothing
- The light source intensity, the properties of the object surfaces and the relative position between the light source and the viewer with respect to the objects affects the formed image in both the camera and the eye
- It's clear that the imaging model must take these factors into consideration

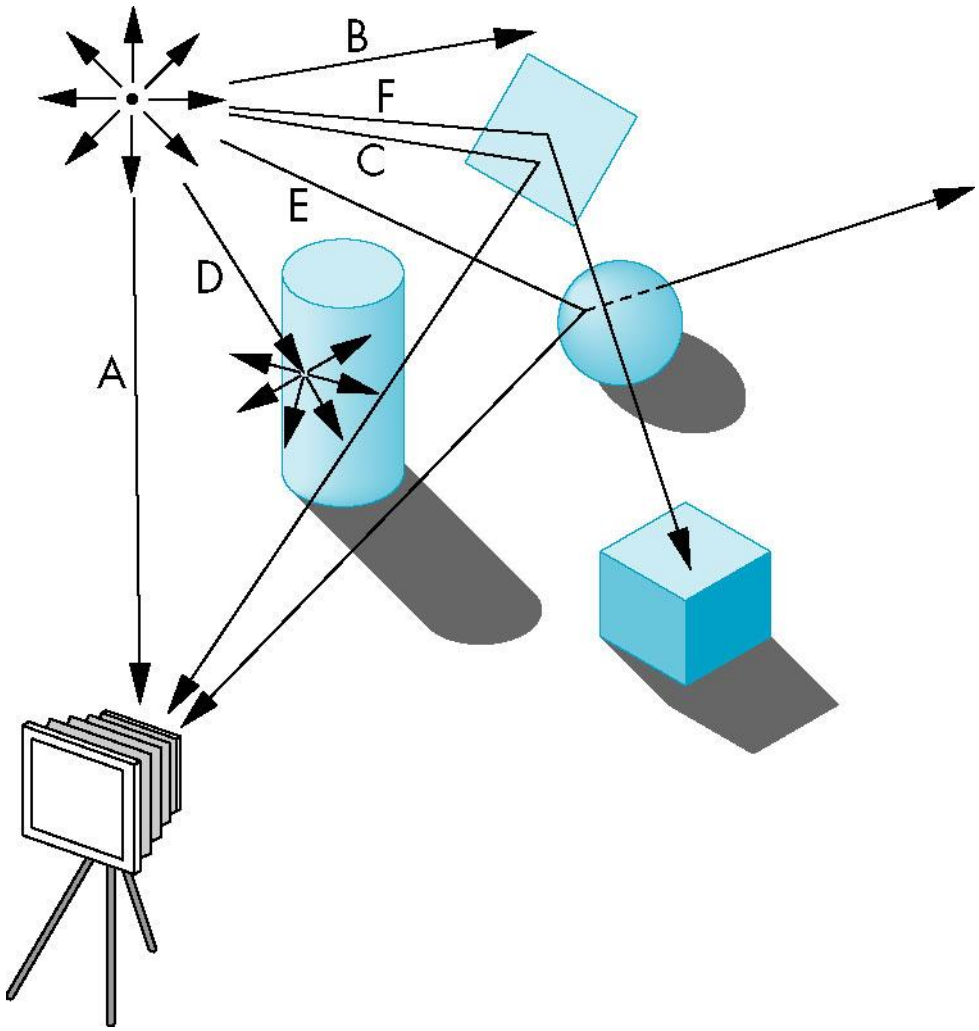




Imaging model

We are interested in a model that can form an image (from a set of objects specifications, the light reflecting properties of these objects, and the properties of the light sources) seen by a viewer with known position and view direction.

- One model is to consider light source as an light rays emitter. The ray travel infinitively in the space
- Some rays can reach from the source directly to the viewer others causes rays to reach the viewer as a result of their interaction with objects
- The properties of the object, lights and positioning determine ray tracing for drawing the image



What we will study?

- Using the imaging model through a graphics library
- We specify the following using function calls
  - Object shapes and positions
  - Object materials and surface properties
  - Lighting
  - Position, direction and viewing space of the viewer
- Then we let the library executes the model, to get the image on the screen
  - The library employs the 3D graphics through imaging model capabilities provided by the GPU (or software emulation if some features are not implemented directly by the GPU)
- Sometimes, we will manipulate the components of the model used (set the shading type, enable or disable operations, etc)